

[illegible]

XTITLE 'EVLIBRARY Symbol Definition Library'

MODULE EVLIBRARY (
LANGUAGE (BLISS32),
IDENT = 'V04-000'
) =

BEGIN

```
*****  
*  
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
* ALL RIGHTS RESERVED.  
*  
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
* TRANSFERRED.  
*  
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
* CORPORATION.  
*  
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
*  
*****
```

++
FACILITY: DECnet Event Logging (EVL)

ABSTRACT:

Event Logging Library of Common Definitions

ENVIRONMENT: VAX/VMS Operating System

AUTHOR: Darrell Duffy , CREATION DATE: 15-June-1980

MODIFIED BY:

V001 TMH0001 Tim Halvorsen 25-Jun-1981
Remove some obsolete definitions

--

%SBTTL 'Definitions'

Structure declarations used for system defined structures to
save typing. These structures are byte sized.
(Thanks to A. Goldstein)

STRUCTURE

BBLOCK [O, P, S, E; N] =
[N]
(BBLOCK+O)<P,S,E>.

BBLOCKVECTOR [I, O, P, S, E; N, BS] =
[N*BS]
((BBLOCKVECTOR+I*BS)+O)<P,S,E>
;

Macro to create a bit id value for net control qio macros

\$BITID

(
Component prefix LNI, NDI, OBI, DLI, ...
Type of parameter V, L, S
Identifier for bit
)

MACRO

\$BITID (COMP, TYP, ID) =
(
(%NAME (COMP, 'SC ', TYP, '_MASK')) ^16 +
(\$BITPOSITION (%NAME (COMP, 'SV_', TYP, '_ ', ID)))
)
%;

EVLIBRARY Symbol Definition Library
Equated Symbols

N 4
15-Sep-1984 23:02:50
15-Sep-1984 22:44:16

VAX-11 BLISS-32 V4.0-742
_S255SDUA28:[EVL.SRC]EVLIBRARY.B32;1 Page (3)

```
0086 0 %SBTTL 'Equated Symbols'
0087 0
0088 0
0089 0
0090 0
0091 0
0092 0
0093 0
0094 0
0095 0
0096 0
0097 0
0098 0
0099 0
```

! EQUATED SYMBOLS:

LITERAL

TRUE	= 1;
FALSE	= 0;
SUCCESS	= 1;
FAILURE	= 0;

!END
!ELUDOM

Version: 'V04-000'

*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*

* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*

* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*

* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*

++

NMAHEAD.B32

Define \$EQLST macro to make library from the NMALIBRY.B32 file

This source is taken from the following source:

--

++

UTLDEF.B32 - UTILITY DEFINITION MACROS FOR BLISS PROCESSING
OF STARLET DEFINITION MACROS.

--

MACRO TO GENERATE EQLST CONSTRUCTS.

MACRO

\$EQLST(P,G,I,S)[A]=
 NAME(P,GET1ST_A) =
 IF NUL2ND_A
 THEN (I) * COUNT*(S) ! ASSUMES I, S ALWAYS GENERATED BY CONVERSION PROGRAM
 ELSE GET2ND_A
 FI

GET1ST_(A,B)=

GET2ND_(A,B)=

B-! KNOWN NON-NULL

: M 0157 0
: 0158 0
: 0159 0
: 0160 0
: 0161 0
: 0162 0

NUL2ND (A,B)=
%NULL(B) %;

End of NMAHEAD

0163 0
0164 0
0165 0
0166 0
0167 0
0168 0
0169 0
0170 0
0171 0
0172 0
0173 0
0174 0
0175 0
0176 0
0177 0
0178 0
0179 0
0180 0
0181 0
0182 0
0183 0
0184 0
0185 0
0186 0
0187 0
0188 0
0189 0
0190 0
0191 0
0192 0
0193 0
0194 0
0195 0
0196 0
0197 0
0198 0
0199 0
0200 0
0201 0
0202 0
0203 0
0204 0
0205 0
0206 0
0207 0
0208 0
0209 0
0210 0
0211 0
0212 0
0213 0
0214 0
0215 0
0216 0
0217 0
0218 0
0219 0

```

:      .TITLE  EVLDEF      Network Event Logger Definitions
:      .IDENT  'V04-000'

*****
*
*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
*  ALL RIGHTS RESERVED.
*
*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
*  TRANSFERRED.
*
*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
*  CORPORATION.
*
*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

++
FACILITY:    DECnet-VAX Network Management Components
              for Event Logging

ABSTRACT:

    Common Definitions for Network Management Event Logging
    These definitions are private to the EVL component.

ENVIRONMENT: VAX/VMS Operating System

AUTHOR:      Darrell Duffy, Tim Halvorsen, 13-June-1980

MODIFIED BY:

    V005      MKP0001      Kathy Perko      27-June-1984
              Now that OPCOM can handle more than 256 bytes, increase
              the length fields for opcom message from a byte to a word.

    V004      TMH0004      Tim Halvorsen     20-Jul-1983
              Increase amount of storage allocated for event
              transmitter NCB.

    V003      TMH0003      Tim Halvorsen     25-Jun-1981
              Add two event flag symbols.

    V002      TMH0002      Tim Halvorsen     20-Nov-1980
              Change definition of second byte of source data

```


General definitions

!...\$EVLDEF

LITERAL

\$EVLST (EVLSC_GBL,0,1

, (SYNCH_EFN,1)

, (ASYNCH_EFN,2)

, (MAXEVTcnt,200)

);

! Event flag used for synchronous I/O

! Event flag used for asynchronous I/O

! Maximum number of events in a queue
! for the transmitter

Processed event record structure

!...\$EVTDEF

MACRO EVTSB_FUNCTION = 0,0,8,0%;

! Function code (= 1)

MACRO EVTSB_FLAGS = 1,0,8,0%;

! Indicates which sinks receive record

MACRO EVTSW_CODE = 2,0,16,0%;

! Event code

MACRO EVTSV_TYPE = 2,0,6,0%;

! Event type within class (see EVCDEF)

LITERAL EVTSM_TYPE = 1^6 - 1^0;

MACRO EVTSV_CLASS = 2,6,9,0%;

! Event class (see EVCDEF)

LITERAL EVTSM_CLASS = 1^15 - 1^6;

MACRO EVTSW_JULIAN = 4,0,16,0%;

! Time: Julian half-days since 1-Jan-77

MACRO EVTSW_SECONDS = 6,0,16,0%;

! Second within half-day

MACRO EVTSW_MSECS = 8,0,16,0%;

! Milliseconds within second

MACRO EVTSW_SRCADR = 10,0,16,0%;

! Source node address

MACRO EVTSB_SRCNAMLEN = 12,0,8,0%;

! Source node name length

MACRO EVTST_SRCNAM = 13,0,8,0%;

! Source node name string (max 6 bytes)

! Event entity follows, type and ID

! Event specific data follows

```

0269 0
0270 0
0271 0
0272 0
0273 0
0274 0
0275 0
0276 0
0277 0
0278 0
0279 0
0280 0
0281 0
0282 0
0283 0
0284 0
0285 0
0286 0
0287 0
0288 0
0289 0
0290 0
0291 0
0292 0
0293 0
0294 0
0295 0
0296 0

```

!
 ! Data block descriptor
 !
 !...\$DBKDEF
 MACRO DBKSL_FL = 0,0,32,0%; ! Forward link in queue
 MACRO DBKSL_BL = 4,0,32,0%; ! Backward link in queue
 MACRO DBKSW_SIZE = 8,0,16,0%; ! Size of structure
 LITERAL DBKSC_SIZE = 10;
 LITERAL DBKSK_SIZE = 10;
 !
 ! Event Queue block
 !
 !...\$EVQDEF
 MACRO EVQSL_FL = 0,0,32,0%; ! Forward link
 MACRO EVQSL_BL = 4,0,32,0%; ! Backward link
 MACRO EVQSW_SIZE = 8,0,16,0%; ! Size of structure
 MACRO EVQSW_EVT_SIZE = 10,0,16,0%; ! Bytes in the event
 MACRO EVQST_EVENT = 12,0,0,0%; ! Start of event data
 LITERAL EVQSC_SIZE = 12;
 LITERAL EVQSK_SIZE = 12;

Structures used in the event transmitter

AST Parameter Control Block

!...\$ASPDEF

MACRO	ASPSL_FL	= 0,0,32,0%;	! Forward link
MACRO	ASPSL_BL	= 4,0,32,0%;	! Backward link
MACRO	ASPSW_SIZE	= 8,0,16,0%;	! Size of structure
MACRO	ASPSW_NETCHAN	= 10,0,16,0%;	! Channel to net device
MACRO	ASPSW_IOSB	= 12,0,16,0%;	! IO status block
MACRO	ASPSW_IOSB1	= 14,0,16,0%;	! Remainder of iosb
MACRO	ASPSL_IOSB2	= 16,0,32,0%;	
MACRO	ASPSL_ROUTINE	= 20,0,32,0%;	! address of routine to perform
MACRO	ASPST_DATA	= 24,0,0,0%;	! Data area address
LITERAL	ASPSC_SIZE	= 24;	
LITERAL	ASPSK_SIZE	= 24;	

Sink control block structure, provides the context for the outgoing logical links from the event transmitter.

!...\$SNKDEF

MACRO	SNKSL_FL	= 0,0,32,0%;	! Forward link
MACRO	SNKSL_BL	= 4,0,32,0%;	! Backward link
MACRO	SNKSW_SIZE	= 8,0,16,0%;	! Size of structure
MACRO	SNKSW_NETCHAN	= 10,0,16,0%;	! Channel to net device
MACRO	SNKSW_IOSB	= 12,0,16,0%;	! IO status block
MACRO	SNKSW_IOSB1	= 14,0,16,0%;	! Remainder of iosb
MACRO	SNKSL_IOSB2	= 16,0,32,0%;	
MACRO	SNKSL_ROUTINE	= 20,0,32,0%;	! address of routine to perform
MACRO	SNKSL_SNKADR	= 24,0,32,0%;	! Address of sink node
MACRO	SNKSL_SRCFL	= 28,0,32,0%;	! Head of source list
MACRO	SNKSL_SRCBL	= 32,0,32,0%;	
MACRO	SNKSL_EVTFL	= 36,0,32,0%;	! Head of event queue
MACRO	SNKSL_EVTBL	= 40,0,32,0%;	
MACRO	SNKSW_EVTcnt	= 44,0,16,0%;	! Number of events on the queue
MACRO	SNKSB_STATUS	= 46,0,8,0%;	! Status of logical link to node
MACRO	SNKSV_STS_OPN	= 46,0,1,0%;	! Link is open
LITERAL	SNKSM_STS_OPN	= 1^1 - 1^0;	
MACRO	SNKSV_STS_BSY	= 46,1,1,0%;	! Some action in progress
LITERAL	SNKSM_STS_BSY	= 1^2 - 1^1;	
MACRO	SNKSV_STS_BKD	= 46,2,1,0%;	! Back door in use
LITERAL	SNKSM_STS_BKD	= 1^3 - 1^2;	
MACRO	SNKSV_STS_DEL	= 46,3,1,0%;	! Delete on close
LITERAL	SNKSM_STS_DEL	= 1^4 - 1^3;	
MACRO	SNKSV_STS_CLS	= 46,4,1,0%;	! Close and delete
LITERAL	SNKSM_STS_CLS	= 1^5 - 1^4;	
MACRO	SNKSV_STS_TMR	= 46,5,1,0%;	! Close on non-use timer outstanding

```

: 0354 0 LITERAL SNKSM_STS_TMR = 1^6 - 1^5;
: 0355 0
: 0356 0 MACRO SNKSB_SNKLOS = 47,0,8,0%; ! Sink mask for lost events
: 0357 0 MACRO SNKSL_SNKLEN = 48,0,32,0%; ! Descriptor of ncb
: 0358 0 MACRO SNKSA_SNKNCB = 52,0,32,0%;
: 0359 0 MACRO SNKST_SNKNCB = 56,0,0,0%; ! NCB of link
: 0360 0 LITERAL SNKSS_SNKNCB = 64;
: 0361 0 LITERAL SNKSC_SIZE = 120;
: 0362 0 LITERAL SNKSK_SIZE = 120;

```

0363 0
0364 0
0365 0
0366 0
0367 0
0368 0
0369 0
0370 0
0371 0
0372 0
0373 0
0374 0
0375 0
0376 0
0377 0
0378 0
0379 0
0380 0
0381 0
0382 0
0383 0
0384 0
0385 0
0386 0
0387 0
0388 0
0389 0
0390 0
0391 0
0392 0
0393 0
0394 0
0395 0
0396 0
0397 0
0398 0
0399 0
0400 0
0401 0

Source descriptor block

!...\$SRCDEF

MACRO SRC\$SL_FL = 0,0,32,0%;
MACRO SRC\$SL_BL = 4,0,32,0%;
MACRO SRC\$SW_SIZE = 8,0,16,0%;
MACRO SRC\$SB-SNKTYPE = 10,0,8,0%;
MACRO SRC\$SB-SRCTYP = 11,0,8,0%;
MACRO SRC\$T-SRCID = 12,0,0,0%;
LITERAL SRC\$S-SRCID = 18;
MACRO SRC\$W-FILTERS = 30,0,16,0%;
MACRO SRC\$T-FILTERS = 32,0,0,0%;
LITERAL SRC\$C-SIZE = 32;
LITERAL SRC\$K-SIZE = 32;

! Forward link
! Backward link
! Size of structure
! Sink type
! Source type code
! Source name
! Number of filters
! Start of filters

Filter descriptor

!...\$FLTDEF

MACRO FLT\$W_CLASS = 0,0,16,0%;
MACRO FLT\$V_CLASS = 0,0,9,0%;
LITERAL FLT\$M_CLASS = 1^9 - 1^0;
MACRO FLT\$V_WLDCOD = 0,14,2,0%;
LITERAL FLT\$M_WLDCOD = 1^16 - 1^14;
MACRO FLT\$Q_TYPESLOG = 4,0,0,0%;
LITERAL FLT\$S_TYPESLOG = 8;
MACRO FLT\$Q_TYPESFIL = 12,0,0,0%;
LITERAL FLT\$S_TYPESFIL = 8;
LITERAL FLT\$C-SIZE = 20;
LITERAL FLT\$K-SIZE = 20;

! Class of event
! Class code
! Wild card code
! Type mask to log
! Type mask to filter

```

0402 0
0403 0
0404 0
0405 0
0406 0
0407 0
0408 0
0409 0
0410 0
0411 0
0412 0
0413 0
0414 0
0415 0
0416 0
0417 0
0418 0
0419 0
0420 0
0421 0
0422 0
0423 0
0424 0
0425 0
0426 0
0427 0
0428 0
0429 0
0430 0
0431 0
0432 0
0433 0
0434 0
0435 0
0436 0
0437 0
0438 0
0439 0
0440 0
0441 0
0442 0
0443 0
0444 0
0445 0
0446 0
0447 0
0448 0
0449 0
0450 0
0451 0
0452 0
0453 0
0454 0
0455 0
0456 0
0457 0
0458 0

```

```

!
! Define structures used by the receiver
!
!
! Define sink type descriptor block
!
!...$SINKDEF
MACRO          SINK$$_LINK      = 0,0,32,0%;      ! Queue links
MACRO          SINK$$_BLINK     = 4,0,32,0%;
MACRO          SINK$$_TYPE      = 8,0,8,0%;        ! Type of sink
LITERAL
$EQUATE (SINK$$_GBL,0,1
, (ACTIVE,254)      ! Active sink types
, (KNOWN,255)       ! Known sink types
, (CONSOLE,1)       ! Console sink
, (FILE,2)          ! File sink
, (MONITOR,3)       ! Monitor process sink
);
MACRO          SINK$$_STATE     = 9,0,8,0%;
LITERAL
$EQUATE (SINK$$_GBL,0,1
, (ON,)             ! Sink is on
, (OFF,)            ! Sink is off, ignore all events
, (HOLD,)           ! Sink is holding all events until turned on
);
MACRO          SINK$$_EVENTS    = 10,0,16,0%;     ! Number of events on queue
MACRO          SINK$$_EVTFL     = 12,0,32,0%;     ! Queue head of event data blocks
MACRO          SINK$$_EVTBL     = 16,0,32,0%;
MACRO          SINK$$_FLAGS     = 20,0,8,0%;      ! Flags
MACRO          SINK$$_DELETE    = 20,0,1,0%;      ! Indicates sink should be deleted when the
LITERAL        SINK$$_DELETE    = 1^1 - 1^0;
! events queued for this sink are output
MACRO          SINK$$_ERROR     = 20,1,1,0%;      ! "error" state! all events are ignored to
LITERAL        SINK$$_ERROR     = 1^2 - 1^1;
! this sink until a data base change
MACRO          SINK$$_MAXBUFSIZ = 22,0,16,0%;     ! Maximum size of buffer (OPCOM monitor only)
MACRO          SINK$$_BUFLN     = 24,0,16,0%;     ! Bytes currently in buffer (OPCOM monitor only)
MACRO          SINK$$_BUFFER    = 26,0,32,0%;     ! Address of buffer (OPCOM monitor only)
MACRO          SINK$$_RAB       = 30,0,32,0%;     ! Address of RAB/FAB storage block (file only)
MACRO          SINK$$_CHANNEL   = 30,0,16,0%;     ! Channel for I/O (monitor only)
MACRO          SINK$$_CLOSERTN  = 34,0,32,0%;     ! Address of routine to perform close
! nonzero if sink has been initialized
MACRO          SINK$$_IOSB      = 38,0,16,0%;     ! I/O status block specific to this sink
MACRO          SINK$$_IOSB1     = 40,0,16,0%;
MACRO          SINK$$_IOSB2     = 42,0,32,0%;
MACRO          SINK$$_NAMELEN   = 46,0,8,0%;      ! Length of sink name string
MACRO          SINK$$_NAME      = 47,0,0,0%;      ! Sink name string
LITERAL        SINK$$_NAME      = 255;
LITERAL        SINK$$_LENGTH    = 302;

```

```

0459 0 LITERAL SINK$K_LENGTH = 302; ! Length of sink descriptor block
0460 0
0461 0
0462 0 !
0463 0 ! Define incoming event channel context block
0464 0 !
0465 0 !...$IECDEF
0466 0
0467 0 MACRO IEC$SL_LINK = 0,0,32,0%; ! Forward link
0468 0 MACRO IEC$SL_BLINK = 4,0,32,0%; ! Backward link
0469 0 MACRO IEC$SW_SIZE = 8,0,16,0%; ! Size of entire structure
0470 0 MACRO IEC$SW_CHAN = 10,0,16,0%; ! Network incoming channel number
0471 0 MACRO IEC$SW_IOSB = 12,0,16,0%; ! I/O status block
0472 0 MACRO IEC$SW_IOSB1 = 14,0,16,0%;
0473 0 MACRO IEC$SL_IOSB2 = 16,0,32,0%;
0474 0 MACRO IEC$B_NCBLEN = 20,0,8,0%; ! Length of NCB
0475 0 MACRO IEC$T_NCB = 21,0,0,0%; ! NCB for incoming link
0476 0 LITERAL IEC$S_NCB = 64;
0477 0 LITERAL IEC$C_MAXNCBLEN = 64;
0478 0 MACRO IEC$T_EVENT = 85,0,0,0%; ! Buffer for event record
0479 0 LITERAL IEC$S_EVENT = 250;
0480 0 LITERAL IEC$C_MAXEVTLEN = 250;
0481 0 LITERAL IEC$C_LENGTH = 335;
0482 0 LITERAL IEC$K_LENGTH = 335; ! Fixed length of structure

```

Define the bits for controlling messages to the batch log
of the event processor.

!...\$ELGDEF

MACRO	ELGSV_DBUPDAT	= 0,0,1,0%;	! Data base updates for transmit or receive
LITERAL	ELGSM_DBUPDAT	= 1^1 - 1^0;	
MACRO	ELGSV_SNKOPN	= 0,1,1,0%;	! Link to sink node opened
LITERAL	ELGSM_SNKOPN	= 1^2 - 1^1;	
MACRO	ELGSV_RCVCCF	= 0,2,1,0%;	! Link confirmed by receiver
LITERAL	ELGSM_RCVCCF	= 1^3 - 1^2;	
MACRO	ELGSV_MONOPN	= 0,3,1,0%;	! Link opened to event monitor
LITERAL	ELGSM_MONOPN	= 1^4 - 1^3;	
MACRO	ELGSV_RAW EVT	= 0,4,1,0%;	! Text of raw event
LITERAL	ELGSM_RAW EVT	= 1^5 - 1^4;	
MACRO	ELGSV_QUEEVT	= 0,5,1,0%;	! Text of event queued to sink
LITERAL	ELGSM_QUEEVT	= 1^6 - 1^5;	
MACRO	ELGSV_RCVEVT	= 0,6,1,0%;	! Text of event received by receiver
LITERAL	ELGSM_RCVEVT	= 1^7 - 1^6;	

Counter descriptor list entry

!...\$CTBDEF

MACRO	CTBSW_PCODE	= 0,0,16,0%;	! Parameter code for counter
MACRO	CTBSW_OFFSET	= 2,0,16,0%;	! Offset in counter block
MACRO	CTBSB_WIDTH	= 4,0,8,0%;	! Width of counter in bits
MACRO	CTBSB_ADDQ	= 5,0,8,0%;	! True for accumulate counter
MACRO	CTBSW_BITMAP	= 6,0,16,0%;	! Bitmap mask for this counter
LITERAL	CTBSC_SIZE	= 8;	
LITERAL	CTBSK_SIZE	= 8;	! Total size of structure

Line id conversion table entry

!...\$VDLDEF

! VMS to DNA Line table

MACRO	VDLSA_VMS	= 0,0,32,0%;	! Address of vms name counted string
MACRO	VDLSA_DNA	= 4,0,32,0%;	! Address of dna name counted string
MACRO	VDLSB_TYP	= 8,0,8,0%;	! Type mask for

```

0540 0 MACRO VDL$V_PNT = 8,0,1,0%; ! point to point lines
0541 0 LITERAL VDL$M_PNT = 1^1 - 1^0;
0542 0 MACRO VDL$V_MUX = 8,1,1,0%; ! multiplexed lines
0543 0 LITERAL VDL$M_MUX = 1^2 - 1^1;
0544 0 MACRO VDL$V_MPT = 8,2,1,0%; ! multipoint lines
0545 0 LITERAL VDL$M_MPT = 1^3 - 1^2;
0546 0
0547 0 MACRO VDL$B_COEF = 9,0,8,0%; ! Unit/tributary coefficient
0548 0 : Unit = vms unit / coef
0549 0 : trib = vms unit mod coef
0550 0 LITERAL VDL$C_SIZE = 10;
0551 0 LITERAL VDL$K_SIZE = 10; ! size of structure
0552 0
0553 0
0554 0 :
0555 0 : IOSB fields
0556 0 :
0557 0 :
0558 0 :!...$IOSBDEF
0559 0 :
0560 0 MACRO IOSB$W_STS = 0,0,16,0%; ! Primary status
0561 0 MACRO IOSB$W_CNT = 2,0,16,0%; ! Normally size of transfer
0562 0 MACRO IOSB$W_STS2 = 4,0,16,0%; ! Secondary status
0563 0 MACRO IOSB$W_STS3 = 6,0,16,0%; ! Tertiary status
0564 0 LITERAL IOSB$C_SIZE = 8;
0565 0 LITERAL IOSB$K_SIZE = 8;
0566 0
0567 0
0568 0 :
0569 0 : End of EVLDEF.MDL
0570 0

```

```

0571 0
0572 0
0573 0
0574 0
0575 0
0576 0
0577 0
0578 0
0579 0
0580 0
0581 0
0582 0
0583 0
0584 0
0585 0
0586 0
0587 0
0588 0
0589 0
0590 0
0591 0
0592 0
0593 0
0594 0
0595 0
0596 0
0597 0
0598 0
0599 0
0600 0
0601 0
0602 0
0603 0
0604 0
0605 0
0606 0
0607 0
0608 0
0609 0
0610 0
0611 0
0612 0
0613 0

```

```

| Version: 'V04-000'

```

```

| *****
| *
| * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
| * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
| * ALL RIGHTS RESERVED.
| *
| * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
| * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
| * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
| * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
| * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
| * TRANSFERRED.
| *
| * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
| * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
| * CORPORATION.
| *
| * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
| * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
| *
| *****

```

```

| ++

```

```

| NMATAIL.B32

```

```

| Source to undeclare the macros required for the precompile of
| NMALIBRY.B32 so they do not appear in the library.

```

```

| --

```

```

| UNDECLARE %QUOTE $EQLST,
| %QUOTE GET1ST_,
| %QUOTE GET2ND_,
| %QUOTE NUL2ND_
| ;

```

```

| End of NMATAIL.B32

```

COMMAND QUALIFIERS

```

| BLISS/LIBRARY=LIB$:EVLIBRARY/LIST=LISS$:EVLIBRARY SRC$:EVLIBRARY+SRC$:LIBHEAD+LIB$:EVLDEF+SRC$:LIBTAIL

```

```

: Run Time: 00:05.8
: Elapsed Time: 00:10.2
: Lines/CPU Min: 6308

```

: Lexemes/CPU-Min: 32531
: Memory Used: 46 pages
: Library Precompilation Complete

0156	0157	0158	0159	0160	0161	0162	0163	0164	0165	0166	0167	0168	0169	0170	0171	0172	0173	0174	0175	0176	0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191	0192	0193	0194	0195	0196	0197	0198	0199	0200	0201	0202	0203	0204	0205	0206	0207	0208	0209	0210	0211	0212	0213	0214	0215	0216	0217	0218	0219	0220	0221	0222	0223	0224	0225	0226	0227	0228	0229	0230	0231	0232	0233	0234	0235	0236	0237	0238	0239	0240	0241	0242	0243	0244	0245	0246	0247	0248	0249	0250	0251	0252	0253	0254	0255	0256	0257	0258	0259	0260	0261	0262	0263	0264	0265	0266	0267	0268	0269	0270	0271	0272	0273	0274	0275	0276	0277	0278	0279	0280	0281	0282	0283	0284	0285	0286	0287	0288	0289	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299	0300	0301	0302	0303	0304	0305	0306	0307	0308	0309	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	0330	0331	0332	0333	0334	0335	0336	0337	0338	0339	0340	0341	0342	0343	0344	0345	0346	0347	0348	0349	0350	0351	0352	0353	0354	0355	0356	0357	0358	0359	0360	0361	0362	0363	0364	0365	0366	0367	0368	0369	0370	0371	0372	0373	0374	0375	0376	0377	0378	0379	0380	0381	0382	0383	0384	0385	0386	0387	0388	0389	0390	0391	0392	0393	0394	0395	0396	0397	0398	0399	0400	0401	0402	0403	0404	0405	0406	0407	0408	0409	0410	0411	0412	0413	0414	0415	0416	0417	0418	0419	0420	0421	0422	0423	0424	0425	0426	0427	0428	0429	0430	0431	0432	0433	0434	0435	0436	0437	0438	0439	0440	0441	0442	0443	0444	0445	0446	0447	0448	0449	0450	0451	0452	0453	0454	0455	0456	0457	0458	0459	0460	0461	0462	0463	0464	0465	0466	0467	0468	0469	0470	0471	0472	0473	0474	0475	0476	0477	0478	0479	0480	0481	0482	0483	0484	0485	0486	0487	0488	0489	0490	0491	0492	0493	0494	0495	0496	0497	0498	0499	0500	0501	0502	0503	0504	0505	0506	0507	0508	0509	0510	0511	0512	0513	0514	0515	0516	0517	0518	0519	0520	0521	0522	0523	0524	0525	0526	0527	0528	0529	0530	0531	0532	0533	0534	0535	0536	0537	0538	0539	0540	0541	0542	0543	0544	0545	0546	0547	0548	0549	0550	0551	0552	0553	0554	0555	0556	0557	0558	0559	0560	0561	0562	0563	0564	0565	0566	0567	0568	0569	0570	0571	0572	0573	0574	0575	0576	0577	0578	0579	0580	0581	0582	0583	0584	0585	0586	0587	0588	0589	0590	0591	0592	0593	0594	0595	0596	0597	0598	0599	0600	0601	0602	0603	0604	0605	0606	0607	0608	0609	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	0620	0621	0622	0623	0624	0625	0626	0627	0628	0629	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	0650	0651	0652	0653	0654	0655	0656	0657	0658	0659	0660	0661	0662	0663	0664	0665	0666	0667	0668	0669	0670	0671	0672	0673	0674	0675	0676	0677	0678	0679	0680	0681	0682	0683	0684	0685	0686	0687	0688	0689	0690	0691	0692	0693	0694	0695	0696	0697	0698	0699	0700	0701	0702	0703	0704	0705	0706	0707	0708	0709	0710	0711	0712	0713	0714	0715	0716	0717	0718	0719	0720	0721	0722	0723	0724	0725	0726	0727	0728	0729	0730	0731	0732	0733	0734	0735	0736	0737	0738	0739	0740	0741	0742	0743	0744	0745	0746	0747	0748	0749	0750	0751	0752	0753	0754	0755	0756	0757	0758	0759	0760	0761	0762	0763	0764	0765	0766	0767	0768	0769	0770	0771	0772	0773	0774	0775	0776	0777	0778	0779	0780	0781	0782	0783	0784	0785	0786	0787	0788	0789	0790	0791	0792	0793	0794	0795	0796	0797	0798	0799	0800	0801	0802	0803	0804	0805	0806	0807	0808	0809	0810	0811	0812	0813	0814	0815	0816	0817	0818	0819	0820	0821	0822	0823	0824	0825	0826	0827	0828	0829	0830	0831	0832	0833	0834	0835	0836	0837	0838	0839	0840	0841	0842	0843	0844	0845	0846	0847	0848	0849	0850	0851	0852	0853	0854	0855	0856	0857	0858	0859	0860	0861	0862	0863	0864	0865	0866	0867	0868	0869	0870	0871	0872	0873	0874	0875	0876	0877	0878	0879	0880	0881	0882	0883	0884	0885	0886	0887	0888	0889	0890	0891	0892	0893	0894	0895	0896	0897	0898	0899	0900	0901	0902	0903	0904	0905	0906	0907	0908	0909	0910	0911	0912	0913	0914	0915	0916	0917	0918	0919	0920	0921	0922	0923	0924	0925	0926	0927	0928	0929	0930	0931	0932	0933	0934	0935	0936	0937	0938	0939	0940	0941	0942	0943	0944	0945	0946	0947	0948	0949	0950	0951	0952	0953	0954	0955	0956	0957	0958	0959	0960	0961	0962	0963	0964	0965	0966	0967	0968	0969	0970	0971	0972	0973	0974	0975	0976	0977	0978	0979	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	0990	0991	0992	0993	0994	0995	0996	0997	0998	0999	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487	1488	1489	1490	1491	1492	1493	1494	1495	1496	1497	1498	1499	1500	1501	1502	1503	1504	1505	1506	1507	1508	1509	1510	1511	1512	1513	1514	1515	1516	151
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-----